

CS 1632 – DEliverable 3: Web testing with bdd

http://lit-bayou-7912.herokuapp.com/



June 20, 2016

Nathan Anuskiewicz

**Summary**

The biggest challenge I faced when writing my tests was accessing individual web elements which did not have a unique name or ID value. For example, I spent a decent amount of time determining how to access the Parse and Compile buttons in Selenium as all three buttons have the same name of “commit”. The other struggle I encountered was accessing the AST in the parse output. When I would try to get text from the ‘code’ element, it would only return the text from the top portion of the page, or, everything but the AST.

I also wrote up one test that I could not get to succeed. This test was designed to test the ‘Back’ link on the pages displayed after any of the three buttons are clicked. After clicking the link, I tried to assert that the URL that the driver was pointing to was the same as the homepage’s URL. However, the test always failed. To go further into this issue, I printed out the current page source after clicking the link. I discovered that the page source returned was the same as the output page (the page with the ‘Back’ link). Therefore the ‘Back’ link was not taking me back to the homepage in my JUnit tests. When speaking with Professor Laboon regarding this issue, he stated that it is most likely because the ‘Back’ link calls a function which may not be recognized by Selenium. This test has been omitted from my test file.

In the future I expect to face many of the same problems as I have already encountered. Poorly designed or more complex webpages may make it difficult to access specific elements. Functions that are not recognized by Selenium could result in failed test cases. Also, the inclusion of JavaScript in a webpage would likely make it much more difficult to test. Overall I found this assignment to be a very enjoyable introduction to web testing, and would like to learn more about it in the future.